Tin upiti

[1. Upit 2](#_Toc104457027)

[2. Upit 3](#_Toc104457028)

[3. Upit 5](#_Toc104457029)

[4. Upit 7](#_Toc104457030)

## Upit

Svi pacijenti koji su primljeni prije datuma X kod doktora Y a koji još nisu primili terapiju

Doktorica Debeljak (id:208) otišla je na godišnji odmor 30.08.2022, te je bilo potrebno zbrinuti njene pacijente koji su joj dodjeljeni prije odlaska na godišnji odmor, a kojima ona još nije prepisala terapiju.

KOD ZA UPIT:

SELECT pa.\*

FROM pacijent as pa, prijem as pr, doktor as d

WHERE pa.id = pr.id\_pacijent

    AND pr.id\_doktor = d.id

    AND d.id = 208

    AND pr.datum\_prijema < STR\_TO\_DATE('30.08.2022','%d.%m.%Y.')

HAVING pa.id NOT IN (SELECT id\_pacijent FROM terapija);

OPIS UPITA:

Kako bi ostvarili ovaj upit potrebno je povezati pacijente sa datumima kada su zaprimljeni u bolnicu, te sa doktorom koji ih je zaprimio. Tablica **prijem** povezuje ove tri relacije jer ona sadrži id svakog pacijenta, datum prijema i doktora koji je pacijenta primio.

* Najprije smo naveli sve relacije pod naredbu FROM odvajajući ih zarezom i dodjeljujući im novi alias. Time smo ih spojili krartezijevim produktom.
* Nakon toga smo povezali tablice preko id-jeva dodajući uvjete pod naredbu WHERE pri čemu smo naveli da id-jevi iz tablice pacijent moraju odgovarati id-jevima pacijenata iz tablice prijem, te da id-jevi doktora iz tablice prijem moraju odgovarati id-jevima doktora iz tablice doktor. Te uvjete smo lančano povezali naredbom AND
* Također dodali smo još i uvjet da id doktora mora biti 208, što odgovara doktorici Debeljak, te da datum prijema iz tablice prijem mora biti raniji tj. manji od datuma odlaska doktorice na godišnji, koji smo unijeli funkcijom STR\_TO\_DATE koja formatira podatak o datumu iz tipa podatka string u tip podatka date.
* Za kraj smo dodali još jedan uvjet naredbom HAVING gdje smo definirali da se id pacijenta ne smije nalaziti unutar skupa id-jeva iz relacije terapija. Za to smo koristili naredbu NOT IN iza koje slijedi drugi upit koji kao rezultat daje id-jeve svih pacijenata koji su primili terapiju.

REZULTAT:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | ime | prezime | datum\_rodenja | adresa | spol | broj\_telefona | MBO | id\_sos\_kontakt |
| 407 | Ema | Knežević | 1995-10-24 00:00:00 | Busoler 18 | Ž | 0957135987 | 24578946 | 307 |
| 418 | Boris | Vlašić | 1984-07-28 00:00:00 | Puljska cesta 19 | M | 092548796 | 815687459 | 318 |

## Upit

Popis svih pacijenata i ljudi koji su bili u doticaju s njima (doktori, sestre, posjetitelji) na katu broj 2 na kojem je izbila zaraza nakon datuma X

Na katu broj 2 bolnice izbila je zaraza među pacijentima i potrebno je sastaviti popis svih ljudi (uključujući i pacijente) koji su se nakon određenog datuma našli na ovom katu.

KOD ZA UPIT:

CREATE VIEW zarazeni\_pacijenti AS

SELECT p.id,p.ime,p.prezime

FROM pacijent as p, prijem, soba, odjel

WHERE p.id = prijem.id\_pacijent

  AND prijem.id\_soba = soba.id

  AND soba.id\_odjel = odjel.id

  AND kat = 2

  AND datum\_prijema >= STR\_TO\_DATE('24.08.2022','%d.%m.%Y.');

-- popis zarazenih pacijenata

SELECT \*  FROM  zarazeni\_pacijenti

UNION

-- popis svih doktora na temelju zarazenih pacijenata

SELECT DISTINCT doc.id, doc.ime, doc.prezime

FROM doktor as doc, zarazeni\_pacijenti as za , prijem

WHERE za.id = prijem.id\_pacijent

AND doc.id = prijem.id\_doktor

UNION

-- popis svih medicinskih sestara na temelju zarazenih pacijenata

SELECT DISTINCT med.id, med.ime, med.prezime

FROM medicinske\_sestre as med, zarazeni\_pacijenti as za , prijem

WHERE za.id = prijem.id\_pacijent

AND med.id = prijem.id\_medicinske\_sestre

UNION

-- popis svih posjeta pacijentima drugog kata na temelju tablice zarazenih pacijenata

SELECT DISTINCT pos.id, pos.ime, pos.prezime

FROM posjeta as pos, zarazeni\_pacijenti as za

WHERE za.id = pos.id\_pacijent;

OPIS UPITA:

Za ovaj upit najprije je bilo potrebno pronaći sve zaražene pacijente sa drugog kata bolnice nakon zadaog datuma. Za ovaj korak bilo je potrebno povezati relacije **pacijent, prijem, soba i odjel**, budući da relacija prijem povezuje svakog pacijenta sa njegovom sobom, a relacija sobe povezuje svaku sobu sa pripadajućim odjelom, te u tablici odjel nalazimo informaciju na kojem katu bolnice se nalazi koji odjel. Ovime dakle dolazimo do informacije na kojem katu bolnice se nalaze koji pacijenti te kada su primljeni u bolnicu. U sljedećim koracima povezali smo te pacijente sa drugim osobama koje su im dodjeljene kroz tablicu prijem (sestre i doktori) te tablicu posjeta.

* U prvom upitu povezali smo sve relacije **pacijent, prijem, soba i odjel** odvajajući ih zarezom nakon naredbe FROM. Naredbom WHERE povezali smo ih preko pripadajućih id-jeva. Pacijenta i sobu sa prijemom, te sobu i odjel.
* Postavili smo uvjet da se pacijenti moraju nalaziti na 2. katu i da moraju biti primljeni nakon ili na datum 24.08.2022.
* U naredbi SELECT odabrali smo podatke o pacijentima
* Za lakše daljnje korištenje rezultate ovog upita spremili smo u novi pogled zvan zarazeni\_pacijenti postavljanjem naredbe CREATE VIEW zarazeni\_pacijenti AS ispred upita.
* Budući da su pacijenti povezani sa sestrama i doktorima preko relacije prijem, u sljedećem upitu povezali smo tablice doktor, zaraženi\_pacijenti i prijem, također preko id-jeva iza naredbe WHERE te smo iza naredbe SELECT odabrali podatke o doktorima. Također bilo je potrebno dodati i naredbu DISTINCT kako bi eliminirali duplikate, budući da više pacijenata može biti primljeo kod istog doktora.
* Prethodni korak ponovili smo još jednom za novi upit no ovaj puta umjesto tablice doktor povezali smo tablicu medicinske sestre.
* Za kraj povezali smo tablice zaraženi\_pacijenti i posjeta navodeći ih iza naredbe FROM i postavljajući uvjet da id zaraženog pacijenta mora odgovarati id-ju pacijenta iz relacije posjeta. Međusobno smo povezali sve ove upite naredbama UNION i tako došli do konačnog rezultata.

REZULTAT:

|  |  |  |
| --- | --- | --- |
| id | ime | prezime |
| 402 | Lissa | Ivić |
| 403 | Klara | Zenzerović |
| 205 | Jan | Nikolić |
| 207 | Božidarka | Obad |
| 503 | Marko | Marulić |
| 1203 | Issa | Biševac |
| 1204 | Elenora | Delfar |

## Upit

Količine lijekova po nazivima kojima ističe rok trajanja za manje od godinu dana

Odjel nabave obavezan je unaprijed naručiti nove zalihe lijekova kojima ističe rok za manje od godinu dana, stoga je potrebno sastaviti popis i stanje takvih zaliha lijekova u bolnici.

Lijekovi kojima je već istekao rok trajanja su također u evidenciji sve do njihovog zbrinjavanja i odlaganja, no u ovom slučaju nas zanimaju samo lijekovi sa još uvijek valjanim rokom uporabe koji se trenutno koriste u bolnici.

KOD ZA UPIT:

SELECT li.id, li.naziv, st.količina, st.rok\_valjanosti,

(SELECT DATEDIFF(rok\_valjanosti,(SELECT NOW() FROM DUAL))) as dani\_do\_isteka

FROM lijek as li

INNER JOIN stanje\_lijekova as st ON li.id = st.id\_lijek

HAVING dani\_do\_isteka BETWEEN 0 AND 365

ORDER BY dani\_do\_isteka ASC;

OPIS UPITA:

Informacije koje su nam potrebne za ovaj upit su naziv, količina i rok valjanosti, a nalaze se u dvije tablice, lijek i stanje\_lijekova.

* Relaciju lijek navodimo iz naredbe FROM i dodjeljujemo joj alias li
* Spajamo ovu relaciju INNER JOIN-om sa relacijom stanje\_lijekova preko id-jeva lijeka
* Iza naredbe HAVING navodimo uvjet korištenjem riječi BETWEEN da dani do isteka moraju biti između 0 i 365 kako bi eliminirali sve lijekove kojima je rok već istekao i kojima će tek isteći ali za više od godinu dana.
* Listu lijekova poredali smo naredbom ORDER BY dani\_do\_isteka prema rastu naredbom ASC na kraju retka kako bi na početku liste dobili one lijekove kojima rok najranije istječe.
* Za kraj iza naredbe SELECT navodimo podatke id i naziv iz tablice lijek, količina i rok valjanosti iz tablice stanje\_lijekova, te dane do isteka valjanosti koje računamo koreliranim podupitom koristeći funkciju DATEDIFF koja prima dva datuma, a kao rezultat vraća broj dana između njih. Za prvi datum navodimo rok\_valjanosti, a za drugi trenutni datum za vrijeme upita koristeći funkciju NOW koja vraća trenutni datum. Ovaj podupit izvršiti će se za svaku n-torku relacije i rezultat će biti prikazan u stupcu dani\_do\_isteka.

REZULTAT:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | naziv | količina | rok\_valjanosti | dani\_do\_isteka |
| 812 | Meropenem | 8756 | 2022-07-02 | 38 |
| 822 | Haloperidol | 356 | 2022-07-05 | 41 |
| 814 | Gentamycin | 890 | 2022-09-07 | 105 |
| 820 | Heparin | 650 | 2022-09-10 | 108 |
| 804 | Ketamine | 1201 | 2022-09-15 | 113 |
| 824 | Cyclizine | 630 | 2022-10-07 | 135 |
| 805 | Bisoprolol | 610 | 2023-02-16 | 267 |
| 818 | Rivaroxaban | 980 | 2023-03-22 | 301 |
| 813 | Vancomycin | 312 | 2023-04-12 | 322 |

## Upit

Koliko je koji doktor propisao lijekova pacijentima u razdoblju između X i Y i tko su doktori koji su prepisali naviše lijekova pacijentima u tom razdoblju?

KOD ZA UPIT:

CREATE VIEW broj\_izdavanja\_lijeka AS

SELECT doc.id, doc.ime, doc.prezime, COUNT(doc.id) as ukupno\_izdano\_lijekova

FROM pacijent as pa

INNER JOIN prijem as pr ON pa.id = pr.id\_pacijent

INNER JOIN doktor as doc ON doc.id = pr.id\_doktor

INNER JOIN terapija as te ON pa.id = te.id\_pacijent

WHERE  datum\_prijema BETWEEN STR\_TO\_DATE('22.08.2022','%d.%m.%Y.') AND STR\_TO\_DATE('29.08.2022','%d.%m.%Y.')

GROUP BY doc.id

ORDER BY ukupno\_izdano\_lijekova DESC;

-- ukupno po doktoru

SELECT \*

FROM broj\_izdavanja\_lijeka;

-- doktori sa najviše izdanih lijekova

SELECT \*

FROM broj\_izdavanja\_lijeka

WHERE ukupno\_izdano\_lijekova = (SELECT MAX(ukupno\_izdano\_lijekova) FROM broj\_izdavanja\_lijeka);

OPIS UPITA:

Kako bi povezali doktora sa brojem izdanih terapija (svaka pojedina terapija prepisuje samo jedan lijek! ) tj. brojem izdanih lijekova potrebno je povezati tablice pacijent, prijem, doktor, i terapija. Tablica prijem povezuje pacijente i doktore, a tablica terapija povezuje pacijente i lijekove koji su im prepisani. Povezivanjem ovih tablica u konačnici ćemo povezati doktore sa lijekvima.

* Upit krećemo od tablice pacijent koju navodimo nakon naredbe FROM i dodjeljujemo joj kraći alias
* Nakon toga radimo INNER JOIN pacijenta sa prijemom, te doktora sa prijemom preko pripadajućih id-jeva te na kraju povezujemo terapiju sa pacijentom na također isti način.
* Nakon naredbe WHERE naredbom BETWEEN postavljamo uvjet između dva datuma koja unosimo funkcijom STR\_TO\_DATE koja datum tipa string pretvara u tip podatka date nad kojim se mogu vršiti razne operacije.
* Narebom GROUP BY grupiramo tablicu prema id-ju doktora kako bi se mogla izvršiti funkcija prebrojavanja COUNT.
* Konačni rezultat sortiramo naredbom ORDER BY prema ukupnom broju izdanih lijekova u padajućem smjeru ključnom riječi DESC.
* Za kraj iza naredbe SELECT specificiramo podatke koje želimo u rezultatu, a to su id, ime, prezime doktora iz tablice doktor (doc) i ukupno\_izdano\_lijekova dobiveno korištenjem funkcije COUNT kojom smo prebrojali prethodno grupirane id-jeve doktora.
* Cjelokupni upit pohranili smo u pogled broj\_izdavanja\_lijeka naredbom CREATE VIEW broj\_izdavanja\_lijeka AS prije naredbe SELECT.
* Selektiranjem svih atributa iz pogleda broj\_izdavanja\_lijeka dolazimo do broja izdanih lijekova po doktoru
* Kako bi dobili doktore koji su izdali najviše lijekova iza naredbe WHERE postavljamo uvjet da ukupno\_izdano\_lijekova mora biti jednako najvišoj vrijednosti iz tog upita. Tu vrijednost dobivamo korištenjem funkcije MAX u podupitu koja vraća najveću vrijednost stupca ukupno\_izdano\_lijekova.

REZULTATI:

|  |  |  |  |
| --- | --- | --- | --- |
| id | ime | prezime | ukupno\_izdano\_lijekova |
| 200 | Krešimira | Paspalj | 5 |
| 208 | Jelena | Debeljak | 4 |
| 201 | Noris | Grubor | 2 |
| 207 | Božidarka | Obad | 1 |

|  |  |  |  |
| --- | --- | --- | --- |
| id | ime | prezime | ukupno\_izdano\_lijekova |
| 200 | Krešimira | Paspalj | 5 |

## Upit

Kako bi ustvrdili koje kategorije lijekova se u bolnici najviše koriste napravili smo upit koji će prebrojati koliko je pacijenata primilo koju vrstu lijeka uključujući i lijekove koji nisu prepisani niti jednom pacijentu.

KOD ZA UPIT:

CREATE VIEW pacijenti\_po\_lijeku AS

SELECT vrsta as vrsta\_lijeka, COUNT(vrsta) as broj\_pacijenata

FROM pacijent as pa

INNER JOIN terapija as te ON pa.id = te.id\_pacijent

INNER JOIN lijek as li ON li.id = te.id\_lijek

GROUP BY vrsta\_lijeka;

SELECT DISTINCT li.vrsta as vrsta\_lijeka, COALESCE(broj\_pacijenata, 0) as broj\_pacijenata

FROM lijek as li

LEFT JOIN pacijenti\_po\_lijeku ON li.vrsta = pacijenti\_po\_lijeku.vrsta\_lijeka

ORDER BY broj\_pacijenata DESC, vrsta\_lijeka ASC;

OPIS UPITA:

Tablica terapija povezuje svakog pacijenta sa lijekom koji mu je prepisan. Jedan pacijent može imati više terapija, tj. prepisanih lijekova. Budući da tablica terapija sadrži samo id lijeka, a ne i podatke o njemu, na to moramo povezati još i tablicu lijek koja sadrži podatak o vrsti lijeka koji je prepisan.

Potom ćemo vršiti grupiranje i prebrojavanje prema vrsti lijeka te ćemo to spojiti sa relacijom koja sadrži sve vrste lijekova u bolnici kako bi u rezultat uključili i lijekove koji nisu prepisani niti jednom pacijentu.

* Upit započinjemo navođenjem relacije pacijent iza naredbe FROM te mu dodjeljujemo kraći alias pa.
* Naredbama INNER JOIN povezujemo relacije terapija i lijek sa pacijentom prema pripadajućim id-jevima relacija.
* Grupiramo rezultat prema vrsti lijeka koristeći naredbu GROUP BY
* U naredbi SELECT odabiremo stupce prikaza, a to su vrsta lijeka i broj pacijenata dobiven prebrojavanjem stupca vrsta funkcijom COUNT.
* Ovim upitom dobili smo sve vrste lijekova koji su prepisani nekim pacijentima i broj pacijenata. Rezultat smo pohranili u pogled pacijenti\_po\_lijeku naredbom CREATE VIEW pacijenti\_po\_lijeku AS ispred naredbe SELECT
* Kako bi u rezultat uključili i lijekove koji nisu prepisani niti jednom pacijentu postavljamo novi upit.
* Tablica lijek sadrži popis svih lijekova u bolnici, stoga ćemo iz nje selektirati stupac vrsta, no budući da bolnica ima mnogo lijekova različitog naziva no iste vrste, potrebno je koristiti naredbu DISTINCT kako bi se eliminirali duplikati.
* U naredbi FROM navodimo tablicu lijek
* Sada ćemo ovu tablicu koja sadrži vrste svih lijekova, spojiti sa prethodnom tablicom pohranjenom u pogledu pacijenti\_po\_lijeku koristeći naredbu LEFT JOIN, a spajanje ćemo vršiti prema stupcu vrsta
* Za kraj rezultat ćemo sortirati prema broju pacijenata padajuće naredbom DESC, te potom prema vrsti rastuće naredbom ASC.
* Potrebno je u naredbu SELECT još uključiti stupac koji sadrži broj pacijenata, no kako bi izbjegli NULL vrijednosti u rezultatu koristiti ćemo funkciju COALESCE koja će iz zadanog seta odabrati prvi rezultat koji nije jednak NULL. Time će vrste lijekova koje nemaju niti jednog pacijenta biti evidentirane kao 0 umjesto NULL.

REZULTAT:

|  |  |
| --- | --- |
| vrsta\_lijeka | broj\_pacijenata |
| Antibiotik | 7 |
| Anelgetik | 4 |
| Antiaritmik | 2 |
| Antikoagulans | 1 |
| Antiemetik | 0 |
| Sedativ | 0 |